Victorian 6502 User Group Newsletter

KAOS

For People Who Have Got Smart

OSI SYM KIM AIM APPLE UK101 ORANGE

Registered by Australia Post
Publication No. VBG4212

ADDRESS ALL CORRESPONDENCE TO 10 FORBES ST., ESSENDON, VIC. 3040

Vol.3 No.8 May 1983

To those members who have sent programs to the Tape and Disk libraries and for use in the school, please accept our thanks. We appreciate their interest and participation.

The good news from the Essendon Primary school is that the teachers and pupils have generated so much excitment and enthusiasm for the two computers that the club loaned to the school that the parents on the school committee have voted to buy a second-hand SUPERBOARD for the school, and with an old T.V. and cassette recorder they have at the school they will soon have three computers running. Incidently, it's amazing what a bit of idle talk will do! I happened to mention that we will need to find a case for the Superboard and Warren Schaeche very kindly donated his old KAOS case, complete with power supply, to the school. Thanks Warren!

The teacher in charge of the computers has written his first program, congratulations Graeme, and reports that the children are showing little interest in arcade type games but are enjoying the challenge of games that make them think.

The Any Base Converter program which was in KAOS 3/7 was written by Ed Richardson and originally appeared in the #2 OSUG newsletter.

The next meeting will be on Sunday 29th May at 2pm at the Essendon Primary School which is on the corner of Raleigh and Nicholson Streets, Essendon. Please note, the school will not be open till 1pm.

The last date for the acceptance of articles for the June newsletter is the 10th June.

INDEX

MY SUPERBOARD II series 2. Part 2

by John Whitehead

The first mod I did was to replace the existing monitor ROM that's at $F800 to $FFFF with a Dabug monitor EPROM.

For those who don't know, Dabug is an improved system monitor written by David Anear. It has single key BASIC, BASIC line editing using CTRL <; >, A, U and D. Deletes characters when backspacing. Shows correct error message. Switches screen format from 24x24 to 48x12 by toggling CTRL B or with POKE 251,x. An instant screen clear with the RUBOUT key, with PRINT CHR$(127) or JSR $FCD5. A screen hold by pressing CTRL and left SHIFT.

To make the screen hold more usful, I wired up a double pole on/off switch to the CTRL and Left Shift keys. (KAOS July 81 and Aug. 81) Apart from holding a BASIC listing on the screen while I make notes, it will also temporarily HALT any running program that prints to the screen including SAVING and output to a printer.

When Dabug is turned on with CTRL Q it alters the input and output vectors as below.

	Normal	Dabug on
Input vector at $0218,9	$FFBA	$F800
Output vector at $021A,B	$FF69	$FB00

Dabug can be turned off by setting the vectors back to normal with POKEs or pressing BREAK.
The Dabug output routine can be used to give 48x12 screen when using the Extended monitor, Assembler and Word processor by using the folowing :-
JSR FBC8 LDA #01 STA FB JSR FB00 JMP to program.
Don't press CTRL Q as this will turn on the input routine as well.
I have altered all my PRINT type programs to run in 48x12 but this is not always easy. I only use 24x24 for graphics.

Dabug uses zero page memory $E0 to $E8 and $F6 to $FF.
When only the output routine is turned on as above, zero page memory $F6 to $F9 and $FC to $FF is used and due to this only three break points can to used with Ext. mon. and Assembler error numbers dont show.

Without Dabug or one of the other improved monitors the Superboard is not convenient to use.

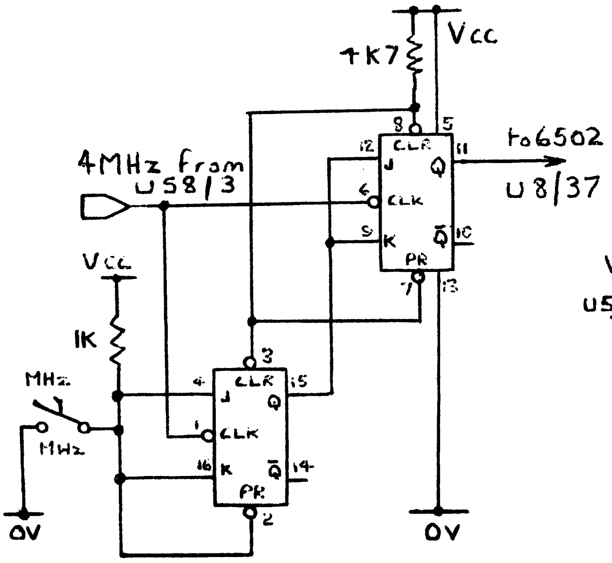
To complete the Dabug mod, I marked all the Key fronts with White 10pt UNIVERS 57 Letraset to show the single key BASIC words. I used a U shaped piece of metal with the ends bent at 90deg to hook under the keytops and pull them off. The letraset was protected with varnish painted on with a brush.

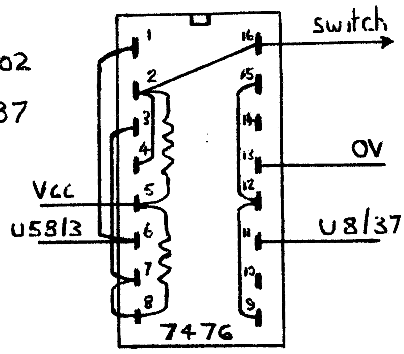
My second job was putting the Superboard and it's power supply in a KAOS designed case. I made a good job of fitting the mains transformer in the case only to find that it affected the TV and I had to take it out and turn it around 90 deg.

The next mod I did was a 1MHz/2MHz switch using the circuit shown.

This was originally in KAOS Feb. 81 and included an 8 input nand with it's inputs connected to the chip select pins of slow chips to automatically switch back to 1MHz. and is used on the C4 505 board.

The 7476 is a JK flipflop and it's output changes state on a negative going clock pulse if PR, CLR, J and K are all high. The switch can be operated while a program is running. When switched to 2MHz all programs run twice as fast, including Dabug line editing. The 2 MHz option is needed to load 600 baud BASIC tapes that don't have nulls after the line feed.





A normal BASIC line saved on tape has a CR after the last character, then 10 nulls (character ASCII 0), then a LF. Next comes the 1st character of the next line. If the NULL statement is used this puts NULLs between the LF and the 1st character. The NULL command will only accept from 0 to 8 Nulls. If more than 8 Nulls are needed use POKE 13,X. Extra nulls may be needed when using a slow printer, to give time for it to do a carriage return.

While on the subject of line composition, below is the format used for machine code saved on tape in checksum. This format is used by the Extended monitor to Save and Load other programs and is used by most M/C programs available on tape. Checksum format is very reliable as it is not possible to load a faulty byte because each line is checked and an Error message comes up, and Loading stopped, if it is incorrect.

Due to the fact that extra error checking characters are required and that every byte is 2 characters, this format is slow to load, but it will work OK at 600 baud or even 1200 baud at 2MHz if you have a good mono recorder.

C H E C K S U M   F O R M A T

Here is the breakdown of a line. All data is in hexadecimal.

Number of data bytes in this line. 14 shown, normally 18

Data bytes — Carriage return — Line feed

;14030011223344556677889 9AABBCCDDEEFF0011223344 08B90D00000000 0A;1803

Address of where the first data byte in this line goes — Check sum — 4 nulls. normally dec. 10 — Start next line

Start of a new line character.

CR, nulls and LF are not displayed.

Superboard

MAY, 1983

Newsletter of the Ohio Superboard User Group, 146 York Street, Nundah, 4012.

SNAKE TAILS by John Froggatt.

Snake Tails is a simple graphics game in which a player steers a snake around the screen using keys E, Space Bar, >, and <. On starting, you use any one of the direction keys to hatch the snake from an egg in the centre of a blank screen. Should you still have a cursor remaining on the screen, try changing 54149 in line 90 to 54118. A number will appear at a random location on the screen and you steer your snake to eat it. When you do, the tail will get longer. The aim of the game is to score as many numbers as possible, and should you drive the snake off the top or bottom of the screen, or allow it to make contact with it's tail, the score is printed and the game ends. The game gets quite difficult once the tail exceeds the screen width.

```
10 REM SNAKE TAILS BY JOHN FROGGATT E=UP <SPC>=DOWN <=LEFT >=RGT
15 DIM A(4),B(4),K(4),L(4):READ A,CS,A1,A2,A3,A4
20 FOR J=1 TO 4:READ K(J),L(J),A(J),B(J):NEXT:POKE 530,0
25 K=57088:T=A:D=0:C=226:FOR J=1 TO 32:PRINT:NEXT:POKE CS,32
30 FORJ=1TO4:POKEK,K(J):IFPEEK(K)=L(J)THEND=A(J):C=B(J):GOTO40
35 NEXT J
40 POKE A,C:A=A+D:IF D<>0 THEN L=1:IF PEEK(A)<>32 GOTO80
45 IF L=0 GOTO 30
50 IF S>0 THEN S=S-1:GOTO 30
55 E=PEEK(T):FOR J=1 TO 4:IF E<>B(J) THEN NEXT
60 POKE T,32:T=T+A(J):L=0:IF Z=1 GOTO30
65 X=INT(RND(1)*A1):Y=INT(RND(1)*A2):A5=A4+X+(A3*Y)
70 IF PEEK(A5)<>32 GOTO 65
75 POKE A5,INT(RND(1)*9)+49:Z=1:GOTO 30
80 H=PEEK(A):IF H<32 OR H>60 THEN PRINT"SCORE";SC:POKE530,0:END
85 S=PEEK(A)-48:SC=SC+S:Z=0:GOTO 45
90 DATA 53776,54149,24,24,32,53381
95 DATA 223,127,1,18,239,191,-32,16,253,239,32,20,251,253,-1,22
```

If you have a C2 or C4 with standard keyboard, then substitute the two data lines below.

```
90 DATA 54304,55104,64,32,64,53248
95 DATA 32,128,1,18,16,64,-64,16,2,16,64,20,4,2,-1,22
```

For a converted C1 with 64 character screen but C1 keyboard, use LINE 95 :-

```
95 DATA 223,127,1,18,239,191,-64,16,253,239,64,20,251,253,-1,22
```

Finally you can substitute a M/C screen clear in line 25. PRINT CHR$(127) for Dabug or PRINT CHR$(26) for Cegmon. The cursor position, 54149 in line 90 will need to be changed to 53413 with a Cegmon screen clear.

Hardware Review - 16K CMOS RAM BOARD FOR C1P

If you intend to expand to disk, the 610 or OZI RABBLE board is definitely the way to go. A 610 would only be considered if the price was much lower than the RABBLE, say $200 or less, because the RABBLE is a better board. If you own a C2 or C4, then the 48k STUDIOTECH board, which plugs into the 48 pin bus, is the best way to expand.

Most of us, however, will probably stick with a C1P in either 24 or 64 character screens, and never aspire to disk status. I, for one, have no intention of ever expanding to disk via a C1P. For me, expansion via a RABBLE board would seem to be a gross waste of the RABBLE facilities.

However I definitely need more than 8k, and would have had to rebuild my power supply even to add 8k the piggyback way. Recently, I purchased a very well made, single sided 16k CMOS RAM expansion board from Bert Patterson. This gives me a total of 24k which is enough for a cassette based system.

The board comes with complete and easy to follow wiring instructions, including overlays of the completed board and the C1P also. The number of chips used is the minimum possible chip count which makes for a small neat board. The board connects to the Superboard via the 40 pin expansion socket, and is fully buffered. It will run at 2MHz CPU clock.

Fully populated, the board draws a miniscule 70 mA of power. With this and the Eprom Extender board (reviewed in a previous KAOS), I have all the Superboard that I want, and very cheaply at that!

The bare board costs $20, or $25 drilled (well worth the extra $5), and add $2 for packing and postage. Address is: Bert Patterson,
and turnaround is speedy.

February 1983 Programming Competition

With more than 450 KAOS members, and lots of schools in the club, I had expected many more than the 7 entries received from 6 entrants. The categories were mixed which made it much harder to vote. Where are the programmers ???

$25 WINNER Alan Calvert with MATHS DRILLS
$10 WINNER Craig (Matt) Dillon with GUNFIGHT AT OK CORRAL
$ 5 WINNER Paul Brodie with THE CHALICE

Next year, probably around January, I will run another competition, so now is the time to get the thinking caps on, or to dig out that part finished program and tidy it up. *TEMPUS FUGIT*

- -

Software review, C4P Golf.

Although this a one person against the course game, it can be enjoyed by several players taking turns, each scoring his own card.

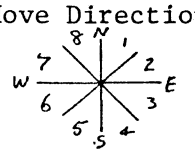
The course is always the same for each game, and is well laid out for the nine holes. With a par of 35, it is rather difficult to get near to this figure on the first few games.

Like most games meant for colour, it loses a little in the presentation when viewed on a black and white monitor, especially the water hazards.

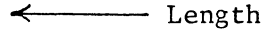
The program lacks details of moves and what clubs to use , and this makes the game hard to play. No paperwork came with the tape. I have included a brief summary below.

Overall, a game very different from the normal shoot-em-down type of arcade program. Definitely for the "Norms" who would like to play golf from an armchair in front of the box. Golf is in the OSUG Library.

Move Directions



Power of Shot(Club)



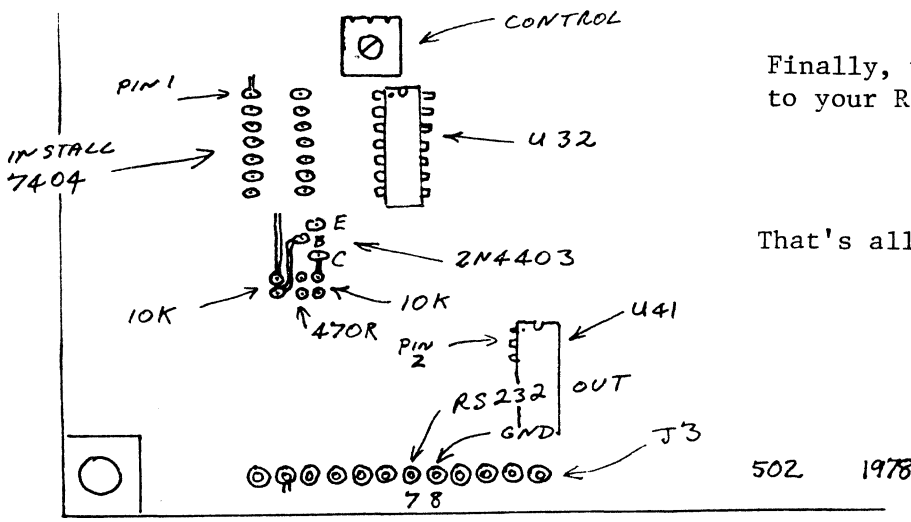
1,2,3,4,5

— SUPERBOARD —

RS-232 for C4P by Bob Best.

The diagram below shows the area where the modification needs to be done on the 502 board of a C4P to give RS-232 output for a modem or printer. Install a 14 pin socket at the U31 position shown. Carefully solder in the 2N4403 transistor (virtually any PNP small signal transistor will do) and the two 10k resistors and the 470 ohm resistor. All resistors can be ¼ watt.
On the pins of the plug to the mother board (J1) wire from pin 24 to pin 27. This provides +5 volts for the RS-232.

Next locate U41, shown in the drawing below. Cut the track going from pin 2 of U41 to pin 7 of J3 as shown in the drawing.



Finally, wire from J3 to your RS-232 socket.

That's all there is to do!

---

2 MHz Clock for C4P by Bob Best

Locate U5D on the 540 board. Cut the track going to pin 13 near that pin. Fit a SPDT switch at a convenient location on the front panel. Wire from the centre terminal to the cut track. Wire from the 2MHz position to pin 14 of the IC, U5D. Wire from the 1MHz switch position to pin 13 of U5D. If you want to be neat, you can wire via a plug fitted to the PROTO socket on the 540 board. Games run excitingly faster with a 2MHz clock!

Send a 9" x 4" SAE to OSUG if you get into trouble and would like full drawings of the mods for 2 Mhz or RS-232.

Software Review - OSI GREATEST HITS VOLUME 2

This is the second package deal offered by Victory Software. While Volume 1 was mainly aimed at the arcade games enthusiast, this tape will appeal to adventure gamers. A single tape, recorded both sides, is supplied, and all programs will run on C1, C2 or C4 without changes.

The first program is a 3D maze, similar to MINOS but with two important plus factors. The graphics are in M/C so that the screen display changes instantly, and the maze has multiple levels so that you can go up or down to other equally complex mazes. It is very easy to lose oneself. You can set the maze size to suit the time you need to play the game. The name of the program is ROACH TRAP. There are three other programs on the tape and all are 8k adventures of medium difficulty:-

African Escape	: Your plane crashes in the Sahara and your task is to escape from the dark continent.
Moon Base Alpha	: To save the colony, you have to prevent a meteor from crashing into the moon base.
Hospital Adventure	: As a secret agent, you have to assassinate a heavily guarded, hospitalised dictator.

None of the adventures need special garbage collector routines. To assist the novice adventurer, Victory supplies a list of verbs used in the adventures. Average solving time is 6 hours. A complete listing is supplied with all programs.

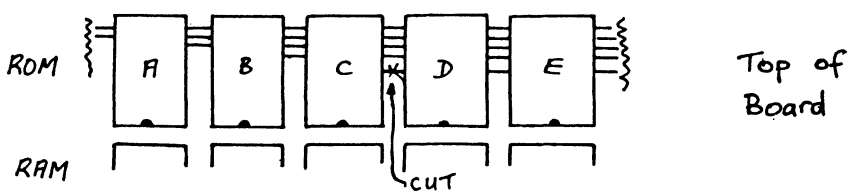
Cost of OSI Greatest Hits, Vol 2 is US$20, which includes packing and air mail postage. When you consider that most adventures sell for around $14-95 each, the package obviously represents very good value for money. Victory accept cheques,money orders, Mastercard or Visa. Address is:-
Victory Software Corp, 7 Valley Brook Road, PAOLI, PA 19301, U.S.A.

Ed Richardson.

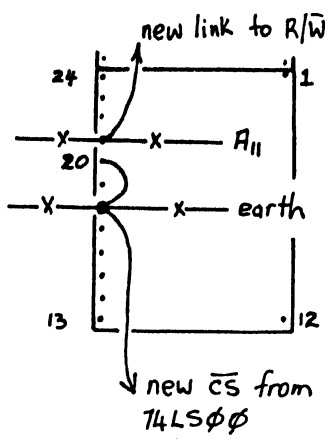
RAM AT $C800 ON RABBLE BOARD

by B. Wills

Adding 2K scratchpad RAM at $C800 on the Rabble board was slightly more involved than shown in the manual. As all ROM and RAM sockets had been soldered in during initial assembly, the chip select to the $CXXX socket was hard to get at. However, it is one of a set of ROM CS signals which form a parallel bus running under the ROM sockets, emanating from U13 (74LS138) near the $FXXX ROM socket. One such track disappears under the $CXXX socket, and it may be cut between the $DXXX and $CXXX sockets. This signal normally goes to pin 20 of the ROM.



The figure on P 6-2 of the manual should be corrected to show the new CS going to pins 18 and 20 (not 17 and 19), and should also show the isolation of pin 18, which is earthed for ROM but not for 6116 RAM. (the diagram below shows board underside)



Isolate pin 18, then connect pin 18 of B socket to pin 18 D socket.
Connect pin 20 to pin 18, connecting both these to the output of 74LS00.
Isolate pin 21, then connect pin 21 of B socket to pin 21 of D socket. Connect pin 21 to pin 21 of a nearby RAM socket. This line is R/W.
The chip select for $CXXX which is used as an input to the 74LS00 can be found as a through-hole under the $FXXX ROM socket, where it is easily soldered.

After some experimenting using 8T28 buffers, including an open collector inverter to generate the DD signal, the system is operating unbuffered ...no 8T28s. It works at 2 MHz with all RAM installed, PIA and VIA and several Eproms, and a video board plugged into the Rabble 40 pin socket.

The original 2114 RAM still occupies 0-8K, with 6116 RAM on the Rabble board from 8-30K.

AN IMPROVED KEYBOARD ALGORITHM

by Rodney Eisfelder

There is nothing wrong with the keyboard of the C-1P/Superboard. However the software used to decode the keyboard is second rate. The keyboard does not behave like either a typewriter or a normal computer terminal.

When shift lock is down, using left or right shift gives unexpected results with most alphabetic keys (shift-K to shift-P are expected to give [\]^_@), but I would hope that (for example) the combination Shift lock, Left Shift and the 'A' key produced an'A' instead of a 'Q'. The keyboard software is even stranger when Shift Lock is in the up position. One peculiarity is that the line feed, Return, Escape and Rub Out keys fail to work. Another strange effect is that the numerical and punctuation keys do not work properly. You must use left shift to obtain the unshifted character and right shift to obtain the shifted character.

We can show the behaviour of the keyboard in a table. The first section of the table gives the important combinations of shift lock, left shift, and right shift (0 means up and 1 means down). We can ignore the case of both left and right shift being down (unless you have three hands). The second part of the table describes the Superboard's keyboard algorithm. The third part of the table describes how a normal keyboard should behave (with allowance for [\]^_@).

SLk	LS	RS	AL	NO	SP	AL	NO	SP
0	0	0	1c	X	X	1c	NM	NM
0	0	1	PX	SH	X	UP	SH	NM
0	1	0	UC	NM	NM	UC	SH	NM
1	0	0	UC	NM	NM	UC	NM	NM
1	0	1	PX	SH	X	UP	SH	NM
1	1	0	PX	SH	X	UP	SH	NM

The abbreviations need explanation:

SLk means Shift Lock
LS means left shift
RS means right shift
AL means the alphabetic keys 'A' to 'Z'.
NO means the numerical and punctuation keys (0-9 and ';,:-./').
SP means the 'special keys' ESC, Line-Feed, Return, Rubout and Space.
1c means lower case letters ('a' to 'z')
UC means upper case letters ('A' to 'Z')
UP means upper case letters for 'A' to 'J' and 'Q' to 'Z' and the special characters [\]^_@
NM means the lower (or only) character on the key
SH means the shifted character on the key
X means an unexpected character (not written on the key top)
PX means an unexpected character apart from [\]^_@

I have succeeded in implementing a very close approximation of the third part of the table that is a few bytes shorter than the existing routine and could be ROMed in place of the existing routine. As a close study of the table shows, this keyboard routine behaves like a computer terminal not a typewriter because only alphabetic keys are affected by shift lock. (However the escape key behaves the same way as in the current routine . Escape only works with shift lock down.)

In the new routine there is no difference between left shift and right shift apart from a necessary quirk to enable access to [\]^_@ when shift lock is up. Although the table above does not mention it, the ConTRol key will only affect alphabetic characters (and is not affected by shift lock). This has a

minor side affect in that ConTRol-'<' and ConTRol-'>' stop working under DABUG. This is easily fixed by changing DABUG to recognize two other characters for cursor positioning (ConTRol-L and ConTRol-R maybe?).

What about WP6502 I hear you ask? Rest assured that the new keyboard routine works with at least some versions of WP6502. Next month, if I get the time I will give a listing of the new, improved keyboard routine. Maybe someone can ROM it.

THE MEETING WAS KAOS

by King Corky

I think I shall have to ease-up on the Rabble Bored, the meetings are getting too quiet.

Now for today's news....George has put together a C3/C8 lookalike with two 8" lookalike double-sided drives, 48K, power supply etc. in a package you can carry under one (scrawny) armpit. The serial terminal can be carried by your resident 'dogsbody'. If our esteemed leader, DJA, had one of these, we wouldn't have to put up with all the grunts, groans & cussing we hear every month as he lugs his, or rather OSI's version of same, up and down the stairs

George also displayed an 8" floppy case that can store up to 90 disks upright, with labelled dividers and a lockable lid. 5 1/4" are also available & both in 45 or 90 disk versions. The cost, well let's just say that if he sold 10 per week for 10 weeks, he would be a very wealthy man!! I suppose we can't really complain about prices for goods from COMP-SOFT, they're still the cheapest anywhere, even without the special deals given to members. For instance, he has just had delivered a stock of the new Alpha-80 printer that can do some amazing things on paper, for less than $680, (which reminds me, I now have for sale a Microline 82A for $850, hardly used.)

Bill Chilcott & Ray Gardiner displayed their, not quite completed, system and it's starting to look very impressive, even though not all software will be fully compatible. The difference will probably not be worth sweating over and GT-BUG looks rather good, from the snippet I saw. Ray also has a FORTH routine that will drive a Votrax speech synth chip, another that will disassemble any micro, (presently set up for 65C02), and modified the editor to put the screen being edited into a window.

Nigel Bisset in Sydney and David, (at considerable cost in phone bills), have developed an intelligent modem program that will allow the remote user to control the host system for the transfer of data as well as allowing the downloading of programs. If the program can be listed on the screen, then it can be loaded into memory. For more information contact David Anear. He would also like a list of all modem owners that are on-line, or soon will be, so he can do further testing without having to call Sydney.

Again the call for help goes out, this time for members living within a reasonable distance of the 'Eyles paper factory', to help collate and fold the newsletter once a month, the week before the meeting. As always, the need is urgent and ALL VOLUNTEERS can contact Rosemary. Many hands make light work

A note to Nigel de Suleau, it's coming and I haven't forgotten you.

CURSOR CONTROL CONVERSION

by Frank Brown

For those members who convert programs from other computers to OSI disk systems, the chart below may be of some use.

FUNCTION	OS 65D3.3	SYSTEM: PET	SHARP MZ-80K	TANDY TRS-80
CLEAR SCREEN (and HOME CURSOR)	PRINT!(28) or	INVERSE HEART	PRINT INVERSE C	CLS
HOME CURSOR	PRINT!(18) or PRINT&(0,0)	INVERSE S	PRINT INVERSE H	NA
CURSOR UP	PRINT!(12)	INVERSE CROSS	INVERSE	NA
CURSOR DOWN	PRINT!(11) or PRINTCHR$(10)	INVERSE Q	INVERSE	NA
CURSOR LEFT	PRINTCHR$(18)	INVERSE	INVERSE	NA
CURSOR RIGHT	PRINTCHR$(16)	INVERSE	INVERSE	NA
REVERSE ON	PRINT!(31,15)	INVERSE R	NA	NA
REVERSE OFF	PRINT!(31,14)	INVERSE	NA	NA
PRINT AT	PRINT@(X,Y)	POKE226,H POKE245,V SYSxxxx ①	POKE4465,X POKE4466,Y PRINTA$*1	PRINT@(X,Y)
64 TO 32 CHR/LINE SWITCH	PRINT!(21)	NA	NA	PRINTCHR$(23)
32 TO 64 CHR/LINE SWITCH	PRINT!(20)	NA	NA	? ②

NOTES:- 1. On the PET the cursor control routine can be called by SYSxxxx (where xxxx is at a different address for each BASIC - 1 to 4). This is faster than moving the cursor in steps. The same applies to the SHARP MZ-80K.

2. Does anyone have the information to fill this square.

I hope that the above information is of some use. Don't forget to give a copy of your program to the disk library when you have finished converting it.

WANTED

If you look through your newsletters you will notice that there are sometimes blank areas at the bottom of the page. Often it is not practical to start a new article in the space available and as we don't really like wasting space, we would appreciate any short items, hints, kinks or otherwise, to fill these spaces and to stop the editor losing any more hair.

KAOTIC LISTS

by Frank Nicholls

"Frank, do you have a mailing list program we could use for KAOS?" This seemingly innocent query by Ian at a meeting some months ago was the first step on a slippery path resulting in a two-disk suite used for all KAOS records and mailings.

This time I was delighted to have experts briefing me. No more rewrites to cope with the maze of understandings between client and consultant. Mark 1 version of the program showed that once again I had failed to get user requirements straight.

"Can you add this?" "Could we have more space in this field?" So back to the drawing board once more, thanking my lucky stars that we were working in BASIC and not in COBOL or RPG-II.

We put your personal particulars, down to the last disk drive, into a DBMS master file with 800 records of 250 bytes covering 19 fields. The file was kept in alpha order to save sorting and was accessed with a six-letter mnemonic name code using 65U's superb FIND command (with search limited to end of data file and a short algorithm to start search in one of eight alpha blocks).

The continual stream of changes of addresses etc went smoothly and mailers were run off each month. Members rolls and selected lists (suppressing the name of those of you who don't want junk mail) were printed.

Next came a legitimate gripe from Marion that inserting names into the alpha list took so much time that she had to schedule it between soap operas.

Then Ian asked if mailers could be printed in mail centre groupings and sorted by post codes to save a lot of hand sorting and labelling. "Why not? Sure we can do that" (after all miracles only take a little longer).

Record insertion into the alpha list (involving moving all records after the insertion point) had been done using a modified version of the inserter from the DBMS nucleus. I had already re-written all of the sub-routines used in file handling etc for other applications and had found that the inserter had bugs in it too.

So a new version was produced and record insertion dropped to under two minutes. It involved a second DBMS master file as an index to the main file. The new file with 3 fields totalling 12 bytes per record is kept in alpha order and the main file is in the order of record creation with deleted records re-used. A master file was used instead of a key file so that records could be inserted and alterations and deletions effected easily.

Ian's problem with Australia Post meant two more master files, one for mail centre data and the other for sorted mail group-post code records. David and I decided that a sorting algorithm in BASIC would suffice since the file once sorted would only need slight re-sorting to include new additions. Quicksort was the fastest so we chose it. Wrong again! After waiting 2 1/2 hours for the initial sort we found that we had to wait the same time for later sorts.

Careful tests showed that whilst Quicksort is fastest for randomly arranged records, it falls behind a Shell-Metzner sort for records nearly in order. Just as I was about to implement a split-merge Shell sort in BASIC, we got access to a machine code version. Now we wait about three minutes, including the time for reading and rewriting the files.

During these tests I had clear demonstration of the speed increases possible if file read/writes are cut to a minimum. By simple changes to the file read,write,read,write sequence involved, the time associated with this phase was reduced from 20 to 2 minutes.

Finished now? I doubt it. David is already murmering that his printer could be saved if only mail centres with more than two members had slips printed for them. And I have decided that with the machine code sort I could use a hashing coding to obviate record Insertion, substituting a sort whenever an alpha list is needed.....

COMP-DOS NOTES

by Frank Halley

1. Another bug in COMP-DOS 1.2! The code which allocates tracks for new files with the "PUT" or "CREATE" commands doesn't work properly in all cases. eg. With track 19 free but track 20 in use, a command like "CREATE FILE,02" will allocate tracks 19 and 20! Very nasty for track 20, especially if you immediately "ZERO" the new file for data storage. The problem is a subtraction operation which should be done in decimal mode, but which is done in binary. Just beware when leaving track allocation to the DOS.

2. I have just purchased COMP-DOS 1.3 for my 'psuedo' C4 (no colour), and have a few notes on it. First, Frank Brown's letter in KAOS 3/7 about quote marks needed around string data statements. This may appear to be true, but is caused by one line in the BEXEC* program (line 540) which disables the comma and semi-colon separators in input strings (and apparently in data statements as well). The solution is not to edit in a lot of quote marks, but to reset the locations changed by line 540 (with POKE 2972,58:POKE 2976,44). Exactly the same problem will occur with OS65D V3.2 if these locations are altered to 13.

3. The ESC 0 command immediately from the keyboard should select inverse printing (according to the documentation), but it doesn't. It actually has the same effect as PRINT!(1), which selects colour 0, and appears as normal text with the TASAN video board. ESC 9 selects colour 14 (same effect as PRINT!(25)) which appears as 'normal' text.

4. The flashing rate of the cursor can be controlled by poking a value into 13884 (standard value is 16). A smaller value will cause a faster flash rate, but don't make the value too large or you'll end up with a sluggish keyboard response.

5. The disk can be easily made to boot in without having to press the 'REPEAT' key. Call in track 1 (which normally loads to $2A00) to $4A00 and change $4E89- to 4C B0 2E. Resave track 1.

6. My COMP-DOS 1.3 disk contained a lot of data which isn't used - presumably left on the master disk by mistake. The unnecessary information was on track6/sector5, 6/6, 11/2, 11/3, 11/4, 11/5, 11/6, 12/5. If you have two disk drives, I find the old copier program from COMP-DOS 1.2 easier to use, and have put it on track 11 and altered COMP-DOS 1.3 to load it with the "COPY" command. The copier program supplied will work for either single or dual drives, but you have to start copying from track 0, and it crashes after use.

I hope it doesn't sound like I'm being critical of COMP-DOS 1.3 or OS65D V3.3, as I think they are great, and encourage all disk owners to buy a copy.

THE BEGINNING MACHINE LANGUAGE PROGRAMMER....part 11

by David Dodds

This month we will examine the cursor routines in ARTIST.

In BASIC games the position of say an aeroplane moving around the screen would be calculated with simple additions and subtractions. Movement of the cursor in ARTIST is handled in much the same way.

Before looking at the cursor routines it would be beneficial to first look at how the 6502 handles the tasks of addition and subtraction. One point to remember- the 6502 has only 8 bit registers and can therefore only handle values up to $FF (255).

ADDITION: Add instructions are given the mnemonic ADC (ADd with Carry).The carry flag is included in the mnemonic because it always takes a part in the addition. Some microprocessors also allow additions which do not involve the carry. Additions on the 6502 can only be performed using the accumulator and memory (in the immediate mode this means the byte following the opcode). The process is

```
  accumulator contents
+ memory contents
+ carry
  ---------------------
  result in accumulator
  ---------------------
```

At the end of the addition the carry, negative, overflow and zero bits in the status register reflect the state of the accumulator. For example, if A contains $D5 and $200 contains $52 then after the execution of ADC $200:

```
                        if carry clear          if carry set
Accumulator  $D5        11010101                11010101
$200          $52       01010010                01010010
carry                         0                       1
                        --------                --------
                    1+  00100111            1+  00101000
                        --------                --------
```

The value on the left hand end which overflows from the accumulator will set the carry bit. Since the value is not zero then Z will be 0 and as the value is less than $80 the N flag will also be 0.

When performing multiple precision additions the carry becomes the link between successivly higher order bytes. In 8 bit additions or on the low order byte of a mulibyte addition it is necessary for the carry to be cleared to 0 before the addition takes place.

SUBTRACTION: The mnemonic for subtract is SBC (SuBtract with Carry). As with the ADC instruction the carry flag always takes part in the process. The subtract operation is actually performed by adding to the accumulator the sum of the complement of the carry and the complement of the value in memory i.e. to subtract, the 6502 adds a negative value. After the subtraction the result is placed in the accumulator and the status flags are updated to reflect the result as follows:

N=1 if the value was greater than $80
Z=1 if the result is zero
C=1 if the value subtracted was less than or equal to the value originally in the accumulator.
The overflow (V) is also affected.

Note that like the CMP opcodes the operation of the carry is the opposite of what you might expect.

Consider the situation where $1A contains $34, A contains $14 and the operation SBC $1A is performed:

```
                        if carry set              if carry clear
$34 =00110100
$34 complement          11001100                  11001100
carry complement (1-C) -       0                - -       1
                        --------                  --------
                        11001100  <== negative value ==>
                                                  11001011
$14                   + 00010100                + 00010100
                        --------                  --------
                      0+ 11100000               0+ 11011111
                        --------                  --------
```

Results are non zero so Z=0, carry is clear indicating that a borrow was required and as bit 7 (highest bit) is set then N=1 signifying a negative result.

As with the ADC instructions the carry bit becomes the link between successive higher order parts of a multiple precision subtraction. For the first 8 bits of a subtraction it is necessary to first set (not clear) the carry.

The cursor of ARTIST you may recall is controlled by 4 subroutines called from the main loop of the program in response to pressing either the U, D, .(>); or ;(<) key. In general terms the tasks to be performed are:

calculate the new cursor position
check if the position is in valid screen space
update cursor and restore the screen character if valid.

In ARTIST the accumulator is used to pass a value to the addition and subtraction routines used to update the cursor. This value is either 1 or the value in TV.LEN. For the moment we will assign the task of validation and update of the cursor to a new subroutine -UPDATE.
The four cursor routines become:

```
CUR.UP LDA #TV.LEN  CURDWN LDA #TV.LEN  CURLFT LDA #1       CURRYT LDA #1
       JSR SUBT             JSR ADD             JSR SUBT            JSR ADD
       JSR UPDATE           JSR UPDATE          JSR UPDATE          JSR UPDATE
       RTS                  RTS                 RTS                 RTS
```

The addition and subtraction routines are:

```
528 ADD    CLC                  402 SUBT   STA MINUS
530        ADC CURPOS           404        LDA CURPOS
532        STA NEWCUR           406        SEC
534        LDA #0               408        SBC MINUS
536        ADC CURPOS+1         410        STA NEWCUR
538        STA NEWCUR+1         412        LDA CURPOS+1
540 AD1    RTS                  414        SBC #0
                                416        STA NEWCUR+1
                                418        RTS
```

to be continued

DEAR PAUL

Q. Is it possible to fit a parallel port to my Superboard?
A. Yes, there are several possibilities. There is a centronics printer interface available which plugs into the ACIA socket, or you may build a board to plug into the 40 pin expansion socket with a PIA (parallel interface adaptor) or VIA (versatile interface adaptor) on it.

Q. How do I transfer 5 1/4" software to 8" disks?
A. Transferring software from 5 1/4" to 8" (or vice versa) is to say the least, difficult. The easiest way is to get a computer with 5 1/4" disks, and another computer with 8" disks and use their serial ports to transfer software. Alternatively there are several computers around with both 5 1/4" and 8" drives, and one of these could be used.

Q. Are CMOS RAM chips more reliable than the usual 2114s?
A. CMOS RAM chips draw less current (and hence dissipate less heat) and also take up less board space than 2114s.

Q. My boyfriend is locking himself in his computer room every night, and I don't see him any more. How can I reach him in his world of computers?
A. Buy a modem.

Q. What are all the small capacitors on PCBs for, and could they be replaced by one single large one?
A The small capacitors on PCBs are called "filter capacitors", they are placed so that the "noise" on the power rails is minimised near various chips.

TESTING 6502 SYSTEMS

This article was sent to us by Darryl Lock, who saw it in the Energy Control newsletter, who reprinted it from the SYM newsletter, SYM-PHYSIS, who in turn reprinted it from Vol.1 No.3 of Commadore's TECHTOPICS. With a pedigree like that it must by OK!

This is a method of testing 6502 systems that fail to operate, particularly if they 'die' when new ROM/RAM has been added.

You take a 'spare' 6502 chip and bend up the data pins. You wire pins 26, 27, 28, 30 and 32 to pin 8 (+5V) and wire pins 29, 31 and 33 to pin 21 (GND). This causes the 6502 to 'see' EA on data lines ie. a NOP instruction.

When this 'doctored' 6502 is inserted in your computer, it will cycle through all 64K addresses in 128 usec or at a 7.63 Hz rate. You then 'probe' address lines, decoder outputs, chip selects etc. looking for the fault, using a CRO or logic probe.

FOR SALE

EPROM CHARACTER GENERATORS FOR OSI
The modified character sets announced in KAOS Vol.2 No.11 and 12, and featuring game or scientific symbols, are still for sale. The C1 and C4 versions of Sargon II have been modified to use the chess pieces in the Enhanced set, and the C4 version also runs on models with C1 keyboard. (Contact KAOS tape librarian for these details.)

Costs are unchanged: 2716 $12.80 2732 (2 sets) $17.80
Modifications and specials also considered. Orders or enquiries (sase please) to:-
B. Wills,

SUPERBOARD Series II, 8K RAM, DABUG III. Metal case, B&W TV.
$300 ono Contact P.J. Collins

PARALLEL PRINTER WM 2000, 120 CPS, bi-directional, logic seeking Matrix printer, 9X9 dot matrix, 4 char. sizes and user defined chars., char. and dot addressable graphics, takes paper from 2 1/2" to 10" with tractor feed.
Price is $550 plus tax. If 4 or more ordered price is $500 plus tax.
For further information contact Brian Campbell
